This listing of claims will replace all prior versions, and listings, of claims in the application.

Listing of Claims:

1. (Currently Amended) A system for optimizing recovery logging, the system comprising:

a storage device on a calling component machine;

a calling component module on the calling component machine, the calling component module adapted to sending a first message without logging the first message to the storage device, and sending a second message to a called component, and logging ~~the~~ a return message to the storage device when the second message to the called component is sent, wherein logging of the return message is an only forced logging event enabling optimized recovery of incomplete requests in event of a system crash; and

a called component table on the calling component machine for storing information associated with the return message,.

2. (Original) The system of claim 1, further comprising a highest log sequence table for storing a highest log sequence number written to a memory buffer and a highest log sequence number written to the storage device.

3. (Original) The system of claim 1, wherein the called component table comprises a log sequence number associated with the called component for determining a status of the return message.

4. (Original) The system of claim 2, wherein the status of a return message is determined by comparing the highest log sequence number written to memory and the highest log sequence number written to the storage device and a log sequence number associated with the called component.

5. (Original) The system of claim 1, wherein a status of the called component's return message is reset when the message is logged to the storage device.

6. (Original) The system of claim 1, wherein the storage device stores a stable log associated with the calling component.

7. (Original) The system of claim 1, further comprising a storage device for storing a stable log associated with the called component.

8. (Original) The system of claim 1, wherein the calling component module sends at least one message to a plurality of called components.

9. (Original) The system of claim 1, wherein the calling component persists over a system failure.

10. (Original) The system of claim 1, wherein the called component persists over a system failure.

11. (Original) The system of claim 1, wherein the stable log associated with the calling component persists over a system failure.

12. (Original) The system of claim 1, wherein a stable log associated with the called component persists over a system failure.

13. (Original) The system of claim 1, wherein only the last message sent by the called component is guaranteed to be stably stored with the called component.

14. (Original) The system of claim 7, wherein the stable log comprises a component identifier and a message.

15. (Currently Amended) A method of recovery logging comprising:

determining whether a first return message received from a called component has been stably logged on a called component machine, wherein the first return message is received in response to a first message sent from a calling component to the called component, wherein the calling component resides on a calling component machine different than the called component machine;

in response to determining that the first return message has not been stably logged, logging at least the first message to a stable log before a second message is sent to the component, wherein the logging enables recovery of incomplete requests in event of a system crash.

16. (Original) The method of claim 15, wherein determining whether the first return message received from the called component has been stably logged comprises:

in response to determining that an entry for the component is not in a table of information associated with called components, adding an entry for the first message to the table, the entry comprising an identifier for the component, and a log sequence number set to a lowest possible value.

17. (Original) The method of claim 16, further comprising:

in response to determining that the entry for the component is in the table of information associated with called components, comparing a highest stably logged log sequence number with the log sequence number in the table entry.

18. (Currently Amended) The method of claim 17, further comprising:

in response to determining that the highest stably logged log sequence number is greater than the log sequence number in the entry, proceeding to call the component without forcing the ~~log,.~~ log.

19. (Currently Amended) The method of claim 17, further comprising:

in response to determining that the highest stably logged log sequence number is less than the log sequence number in the table entry, forcing the log so that the highest stably logged record has a higher log sequence number than the table ~~entry~~ entry.

20. (Original) The method of claim 19, further comprising updating the highest stably logged log sequence number to the highest log sequence number written to the stable log.

21. (Original) The method of claim 15, further comprising sending at least one message to a plurality of called components before forcing a log.

22. (Original) The method of claim 15, wherein the called component persists over a system failure.

23. (Original) The method of claim 15, wherein the calling component persists over a system failure.

24. (Original) The method of claim 15, wherein messages sent to the called component are uniquely identified.

25. (Currently Amended) A computer-readable storage medium including computer-executable instructions for:

sending a first call message to a called component from a calling component;

sending a second call message to the called component from the calling component;

logging a return message to the first call message in a stable log ~~associated with~~ on the called component machine; and

logging the return message to the first call message in a stable log ~~associated with~~ on the calling component machine when the second call message to the called component is sent.